DOCUMENT RESUME

ED 346 845 IR 015 635

AUTHOR Kay, Robin H.
TITLE Charting Pathways of Conceptual Change in the Use of Computer Software.
PUB DATE Apr 92
NOTE 30p.; Paper presented at the Annual Conference of the American Educational Research Association (San Francisco, CA, April 20-24, 1992). For a related paper, see IR 015 641.
PUB TYPE Reports - Research/Technical (143) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS Adults; *Cognitive Processes; Computer Literacy; Computer Software; Foreign Countries; Graduate Students; Higher Education; Interaction; *Learning Strategies; Metaphors; Prior Learning; Research Methodology; *Spreadsheets; Task Analysis; Videotape Recordings

ABSTRACT

Researchers of computer ability have been largely influenced by the Galtonian perspective of intelligence assessment: a predominantly linear, construct-driven model based on identifying statistically determined factors. The use of this methodology, though, has far more to do with utility than theoretical rigor. Notably absent from this kind of approach is data on process--how a subject interacts with the computer. This study looked at the process of knowledge acquisition in a computer-based environment. Six subjects (two males, four females) were videotaped while learning a new spreadsheet software package. A detailed examination of their think-aloud protocols is presented which addresses the role of previous experience, the use of metaphors, the effects of task interpretation, and use of terminology. The following conclusions were reached: (1) there is no clear relation between previous computer-related skills and the successful completion of spreadsheet learning tasks; (2) subjects actively attempt to learn by using a variety of metaphors; (3) task interpretation affects how subjects behave and the kind of errors they make while learning; (4) terminology is related to degree of understanding of new tasks; and (5) a process-oriented approach to examining computer ability provides a rich source of theoretical and pedagogical information. Protocol instructions for subjects are appended. (22 references) (Author/BBM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED346845

Charting Pathways of Conceptual Change in the Use of Computer Software

Robin H. Kay

Ontario Institute for Studies in Education

Centre for Applied Cognitive Science

252 Bloor St. West (9th floor)

Toronto, Ontario

M5S 1V6, CANADA

(Email: R_KAY@UTOROISE)

IR015635

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Robin H. Kay

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

BEST COPY AVAILABLE

Abstract

Charting Pathways of Conceptual Change in the
Use of Computer Software

Researchers of computer ability have been largely influenced by the Galtonian perspective of intelligence assessment: a predominantly linear, construct-driven model based on identifying statistically determined factors. The use of this methodology, though, has far more to do with utility than theoretical rigour. Notably absent from this kind of approach is data on process--how a subject interacts with the computer. The following study looked at the *process* of knowledge acquisition in a computer-based environment. Six subjects (2 males, 4 females) were videotaped while learning a new spreadsheet software package. A detailed examination of their think-aloud-protocols is presented examining the role of previous experience, the use of metaphors, the effect of task interpretation, and use of terminology. The following conclusions are made: 1) there is no clear relation between previous computer-related skills and the successful completion of spreadsheet learning tasks; 2) subjects actively attempt to learn by using a variety of metaphors, 3) task interpretation effects how subjects behave and the kind of errors they make while learning, 4) terminology is related to degree of understanding of new tasks, and 5) a process-oriented approach to examining computer ability provides a rich source of theoretical and pedagogical information.

Charting Pathways of Conceptual Change in the Use of Computer Software

Defining the nature of computer ability, or computer literacy as it is more commonly referred to, has proven to be an illusive endeavour. Since its relatively modest beginning in 1972 (Moont, 1987), the term ***computer literacy*** has flourished into a myriad of conflicting meanings and purposes leaving a somewhat amorphous result. The phenomenally rapid growth of technology since that time has significantly influenced the design of software, further complicating what it means to "know about computers".

Noticeably absent from the computer ability literature is research on the ***process of learning***--how computer related concepts are acquired and used. Instead investigators have relied solely on a paper-and-pencil format to glean information about what people know about using computers. This relatively straightforward strategy has produced a wealth of well-organized, albeit conflicting, information (Kay, 1989, in press). Yet this information offers little toward understanding the dynamics of human-computer interaction.

A number of investigators in the area of education and cognitive science have argued for more research on the knowledge building process. Chi & Bassok (1989) maintain that there is a need for more explicit examples describing the conditions under which various knowledge building actions apply. Glaser (1989) and

Glaser & Bassok (1990) note that while considerable advances have been made in the areas of memory organization, knowledge required for solving problems, and characteristics of understanding, the knowledge acquisition process has not been extensively examined. *

Several researchers (Bereiter & Scardamalia, 1989; Brown & Palinscar, 1989; Collins, Brown, & Newman, 1989) have identified specific a number of successful knowledge acquisition activities including rehearsal and review, monitoring, unpacking implicit assumptions, summarizing, considering alternatives, questioning, clarifying and predicting. Few have offered a detailed analysis of knowledge building characteristics in a natural setting (Ceci, 1990; Siegler, 1989).

In many respects, the computer milieu is ideally suited for process-oriented research. Most software packages offer a set of limited, clearly identifiable tools which can be used any number of ways by a given user. The interaction between user and computer, therefore, is self-contained and trackable, yet flexible enough to permit considerable variability. Furthermore, the novelty of computers for adults, provides a subject population that is articulate and motivated.

It is time for researchers of computer-related abilities to follow the lead of their peers in education and cognitive science. A shift from categorical, paper-and-pencil methods to process-oriented procedures is required to develop a comprehensive and coherent understanding of how individuals

acquire and apply computer-related concepts. This kind of shift will also help researchers explore individual variability in human-computer interaction, an issue inherently obscured by a categorical approach which tends to focus on universals and consistencies (Keating, in press).

The purpose of the following pilot study is to critically examine the process of knowledge construction in a computer environment. A detailed, anecdotal analysis of six think-aloud-protocols will be looked at in terms of the following components: role of previous experience, use of metaphors, task interpretation, and the use of terminology. This study is preliminary effort to demonstrate the theoretical and pedagogical potential of process-based research on computer ability.

Method

Sample and Measures

The convenience sample consisted of 6 volunteers (3 graduate students and 3 professionals) (2 male, 4 female), ranging from 25 to 31 years of age.

Survey Data. Subjects filled out a brief, Likert-style, questionnaire (7 point scale) on their intentions to use computers and their affective and cognitive attitudes. A detailed interview on the kinds of software they had used and their understanding of this software followed.

Subjects varied in experience and ability: two beginners,

three intermediates, and one advanced user. The following brief descriptions of each subject's ability and experience with respect to using computers were used to help interpret the verbal-protocol data.

Beginner Level:

Subject 1: He was a 30 year old attorney, who did not own a computer. He described himself as having moderate intentions to use computers for practical purposes, but unlikely to experiment or play with computers (23/42 on intentions to use computers). He also felt relatively positive about computers, both in terms of affect (35/42) and cognition (33/42). He had low-to-moderate hardware (27/48) and moderate software (27/36) ability. More specifically, he felt very confident using a word processor (54/60), but had no experience in other software packages.

Subject 2: She was a 25 year old graduate student, who did not own a computer and had used a word processing package, somewhat sporadically, for the past 6 years. She had moderate intentions to use computers (22/42), but relatively positive affective (36/42) and cognitive (35/42) attitudes toward computers. She had low-to-moderate knowledge of hardware (28/48) and moderate knowledge of software (26/36). Her knowledge of word processing skills was high (50/60), but she had only used one other software package on a mainframe computer several years ago for statistics.

Intermediate Level:

Subject 3: She was a 28 year old graduate student who owned a Macintosh computer. She had moderate intentions to use computer (22/42), but relatively positive affective (33/42) and cognitive (36/42) attitudes toward computers. She had moderate knowledge of both hardware (32/48) and software (29/36). She had used a number of Macintosh software packages including word processing, spreadsheets, data base, graphics and statistics. While her knowledge of word processing was high (53/60), her scores for spreadsheet (26/40), database (33/56), and utility (14/32) software were low. She also had no knowledge of programming. She had used computers for 5 years.

Subject 4: He was a 31 year old, market researcher who did not own a computer. He did not complete a questionnaire so the following is a general description of his ability. He described himself as having moderate knowledge of word processing and spreadsheet software. While his knowledge of hardware was relatively weak, he felt his software ability was moderately high. He was very positive about using

computers, particularly with respect to spreadsheet software. He had been using computers for about 2 years. **Subject 5:** She was a 27 year old graduate student who owned IBM clone computer. She only intended to use the computer for practical purposes, but had high cognitive (39/42) and affective (39/42) attitudes toward computers. She completed a shorter version of the computer abilities questionnaire, so detailed scores were unavailable. She had been using mainframe computers for 7 years and microcomputers for the past 3 years. She had strong knowledge of word processing, DOS (operating system), graphics, statistics software. She was familiar with database, programming and utility software.

Advanced Level:
Subject 6: She was a 30 year old research associate who did not own a computer. She also had an M.A. in computer science and a Ph.D. in Mathematics. She only used computers for practical purposes (26/42) and had moderate to high affective (31/42) and cognitive (35/42) attitudes toward computers. She had been using both mainframe and micro computers for 13 years. She had perfect scores on both hardware and software assessment measures. She also had perfect scores in word processing, data base, programming, and DOS software. She had considerable experience with spreadsheets (31/40) and utility software (30/32). She had used over 7 different kinds of computers.

Computer Data. Subjects were videotaped for 60 minutes (the camera was focused on the screen) while learning a common spreadsheet package (Lotus 1-2-3, Version 2.2) on an IBM, 20286 clone. A spreadsheet software package is commonly used by accountants to enter rows and columns of numbers. A similar task could be done using a calculator and a lined sheet of paper. Each subject was asked to do the following tasks: move the cursor, enter rows and columns of numbers, enter labels, insert blank rows and columns, move and/or copy rows or columns of data, and compose formulas to add numbers.

Subjects were given instructions similar to those given in a

typical protocol analysis (Appendix A- from Oatley, 1988). The standard procedure was to introduce the subject to a task and encourage self-directed learning. It was critical to give subjects at least some tasks which they had not done before. Individuals who are too familiar with a task have difficulty articulating what it is they are doing and why because the task is too automatic (Oatley, 1988). Unlike typical protocol analysis (Ericsson & Simon, 1984), subjects were given calculated "hints" when they were unable to proceed. This "question-asking" protocol technique was used with some success by Kato (1986). Every effort was made to encourage subjects get "unstuck" on their own. Attempts were made to limit intervention to situations where subjects did not know syntax of a particular command.

The first 25 minutes of each of the six subject videotaped session was transcribed. Verbal expressions and sounds, as well as critical keystrokes were included in the transcriptions.

Results and Discussion

Four components of the knowledge building process were examined including the role of previous experience, the use of metaphors, task interpretation, and the use of terminology. Conclusions, supporting vignettes, and a discussion of each component will be presented.

Role of Previous Computer-Related Skills

Intuitively, one would predict that previous computer-

related skills, such as typing or use of different kinds of software, would be positively and strongly correlated with successful completion of the spreadsheet learning tasks. Intermediate and advanced users should out perform beginners in terms of time taken to complete a task and ability to grasp new concepts. Yet, in this small sample, computer-related knowledge did not appear to be systematically related to successful progression through a task.

The effects of four kinds of computer related knowledge will be discussed: mathematical and programming knowledge, typing skill, number of software packages learned previously, and knowledge of shortcuts.

Mathematical and programming knowledge: Recall that Subject 6, an advanced level user, had considerable knowledge and interest in both computers and mathematics with graduate degrees in both areas. The format of a formula to add numbers (@SUM(a1..a4) was demonstrated to her and she was asked to use a similar formula in another related task. With a strong background in numbers, formulas, programming, math and computers, and an example in front of her, one would predict successful completion in a relatively short time. The task took her almost 20 minutes (19:51) to complete. Compare this time with Subject 1, a beginner, who completed a similar task in four and a half minutes (4:27) and Subject 2, also a beginner, in just about 5 minutes (5:02). Intermediate levels subjects took 2:59 (Subject 3), 1:41

(Subject 4), and 2:30 (Subject 5).

At the beginning of the task, she quickly began to type in the formula, but forgot to start it off with a critical but somewhat arbitrary "@" sign. Instead of typing @SUM, she typed SUM. Her syntax from that point on, with a few corrections was accurate. She had the correct formula in just over a minute and was told that the formula was correct at the two minute mark. Still she pursued a mathematical solution. She used the software's help information (no other subject did this) and spent considerable time trying to find the answer. In essence, she was taking a fairly complicated line on how to solve her problem, especially since she did have an example (that I had typed in) available to her. When she finally typed in a syntactically, but not substantively (she identified the wrong group of cells to be added), correct formula at 11 minutes, she continued to apply mathematical knowledge to solving her problems attempting to identify rounding errors. Her programming knowledge may also have been a hinderance. A number of errors in numerical programming involve truncation, rounding, or chopping off digits.

Subject 6's extensive knowledge and experience may have inhibited her progress on this moderately difficult spreadsheet task. Ceci (1990) notes that although knowledge can be recruited to help us solve problems, sometimes "it can blind us to more efficient ways" (p. 119). He cites two examples: one where J. von Neumann, a renowned mathematician, performed remarkable

mathematical gymnastics to solve a relatively straightforward high-school physics problem; and the other, where Alan Turing, the mathematical logician, attempted to fix his bike through a fairly lengthy and convoluted procedure. A cycle mechanic could have solved the problem in five minutes (p. 119, Ceci, 1990). What appears to be critical is not only the level of knowledge but the type of knowledge one acquires and the context of knowledge acquisition.

Subject 6's mathematical/programming approach would have worked well in a programming environment, but a supposedly friendly application environment alters the context enough so that relatively straightforward tasks become time consuming and effortful. Subjects 1 and 2, on the other hand, had little knowledge that would interfere with there solving a formula problem. They possessed enough mathematical knowledge to understand algebraic formulas and a need to pay attention to syntactic details. Subject 6 seemed to have taken syntax for granted.

Typing skill. Typing skill would be considered by most people to be an asset for using computers. Subject 6 was able to type in numbers and labels at a much faster pace than the other subjects--she was the only subject with formal typing training. On the other hand, her knowledge of typing prevented her from exploring how the "insert key" worked. She maintained that it was faster for her to type he formula over again than to fiddle with

the insert key. While typing skills may be important enter information, they may not be especially helpful when learning new software. A slow, deliberate pace appeared to be most effective in this study, allowing subjects (1, 2 and 4) to observe the effect of their keystrokes.

Software packages. One might also assume that the more software packages one knows, the easier it would be to learn a new software package. Again this reasonable assumption proved to be too simple to account for subjects' behaviour in this study. Subject 6 had an advanced knowledge of software packages, but also an equally advanced knowledge of error types. When she had a problem with a specific task, she said that there could be a number of possible errors. On several occasions, she methodically prodded through each error type, when she began to experience difficulty. Subjects 1 and 2, though, with only word processing experience, had a small repertoire of possible errors to check--syntax and perhaps misplaced cell locations. They would never consider truncation errors, or look in a help menu, or know that different spreadsheets might have different syntax. Sometimes a little knowledge is better than an abundance of knowledge, particularly for simple problems.

Shortcuts. With a number of software packages, there are long ways of doing tasks and short ways or shortcuts. Beginners in this study were unaware of shortcuts and had to do tasks the long way. Intermediate level subjects 3 and 5 knew of shortcuts

and even asked, on occasion, whether there was a faster way of doing a task. While shortcuts may save time, they can also cause problems when they prevent users from adequately attaining the basic knowledge of an action. Subject 5 used a shortcut in the spreadsheet menu. Instead of moving the cursor over to a command and pressing the enter key, she simply pressed the first letter of the command. She did not have a complete understanding of how the menu worked, and this shortcut inhibited her understanding. She was often confused about where she was in the software package. Subjects 1 and 2, on the other hand, were better able to use the menu line, partially because they were slowed down by using the long way, and therefore were able to pay attention to critical changes and details.

Incidently, after reviewing the videotapes for Subjects 3 and 5, I observed my adapting to their quick pace. On several occasions I showed them shortcuts because I perceived they were frustrated at the slow pace. In retrospect, my suggestions were probably detrimental in that they prevented these subjects from slowing down and observing the cause-and-effect of their keystrokes.

The relation among specific computer skills and learning new skills is complex. The survey data in this study would not have been able to account for time taken to complete the formula-learning task. The protocol data revealed a number of specific instances where previous knowledge was more detrimental than

helpful. While it is often assumed by novices that using computers requires the ability to program and type, it has been argued (Kay, 1989) that these skills are somewhat antiquated in terms of today's software. The protocol data here is consistent with this position.

The Use of Metaphors

In this study, an assortment of metaphors were used by subjects to complete and learn new tasks on the computer. These metaphors were used extensively throughout the learning protocol. In some instances, the metaphors were helpful; in other instances, they hampered a subject's progress. On certain occasions, subjects clung tenaciously to an old metaphor in spite of contradicting evidence. This finding has also been reported by Kuhn (1989) in the acquisition of scientific concepts. Individuals hold steadfastly to their metaphors until presented with complete and improved versions.

Subjects in this study, borrowed metaphors from previous knowledge of word processing, icon or menu driven software (e.g. Macintosh type software), banking machines, graphs, calculators and game playing.

Word processing metaphor. Word processing knowledge was often used to solve spreadsheet problems, particularly with respect to editing. Tasks such as entering numbers, moving and copying text were often attempted with a word processing metaphor in mind. For example, all subjects except for the advanced user

hesitated after typing their first number into the spreadsheet. Spreadsheet software requires that you press enter, and until the enter key is pressed the number appears and remains in the top left corner of the screen. This idiosyncrasy particulary surprised beginners who claimed the procedure was different for word processing. Subjects 1 and 2 (beginners) openly claimed that knowledge of copying and moving helped in understanding the same process in the spreadsheet software. Subject 6 (advanced user) was able to quickly use editing keys that were common to both word processing and spreadsheets.

Icon/Menu metaphor. A metaphor based on icon or menu driven software, commonly associated with Macintosh computers, appeared to be the most troublesome for subjects when attempting to use it with spreadsheet software. The most common difficulty was experienced when attempting to move or copy text. Subjects 3 and 5 wanted to block off text (as if they were using a mouse) when trying to transfer numbers from one location to another. Subject 2, 3, and 5 attempted to use cursor keys to move vertically in the spreadsheet menu (the enter key was the correct choice here).

Subject 3, who owned a Macintosh computer, was so entrenched in icon/menu" metaphor, that she repeatedly expressed animosity toward the software. She noted from the start that the software looked "pretty boring" and that the opening screen was "telling her nothing" (other subjects (1, 4, and 6) noted helpful details from the same screen). She regularly commented that most

operations seemed excessively cumbersome. While this attitude prompted her to seek more efficient solutions to some problems, it also inhibited and restricted her observation of the cause-and-effect of certain keystrokes. When I introduced the spreadsheet menu she was clearly happy, exclaiming "Yahoo", but when the menu did not work like a Macintosh menu, she retreated into her critical and evaluative position.

While subject 3's fixation on the "Mac" way of doing things is only one example, it seems reasonable to assume that individuals learn a variety of software packages on a variety of computers, and that a certain degree of fixation occurs. This is tangentially related to the broader issue of theory fixation. Sternberg (1989) argues that knowing too much (or using something to long) produces a deep structure that can seriously impair functioning. Moving beyond entrenched ways of doing things, particularly in the computer environment where change is the operative word, is a serious challenge for educators.

Banking machine metaphor. A final example shows that metaphors need not be complicated to be useful in learning new software. Three subjects (1, 5 and 6) had considerable difficulty with moving and/or copying text. While one subject was a beginner, the other two had extensive computer knowledge and one would expect that they would complete the moving/copying task with relative ease. Problematic metaphors included the desire to block text with a mouse (mentioned earlier), expecting the use of

buffers (commonly used in programming), and excessive weight placed on the position of the cursor. I came to expect subjects to have difficulty with this task. Subject 2, a beginner, completely the task quickly and without error. Given her limited experience, I thought this was remarkable. After some thought, she said "it was like using a bank machine where you transfer money from one account to another".

Other metaphors. Other metaphors used were based on knowledge of graphs, calculators, and games. Subject 2 described the opening screen as an inverted graph--she quickly adjusted her metaphor when she saw the cursor move, labelling the screen more appropriately as "a grid". Subject 2, when asked to add numbers, claimed that he was looking for something on the keyboard that was similar to a calculator. He also said that one way of doing the task would be to add the numbers with a calculator and enter the sum into the spreadsheet. This would clearly undermine the power of using spreadsheet software and is probably also based on how one would do the same task on a word processor. Subject 4, after experiencing some difficulty using fancy formatting commands, claimed that he was sure there was some "trick" to doing the task, as there might be in a game. While computers often appear to be "tricking" unsuspecting users, in this instance the "trick" metaphor prevented subject 5 from noticing a simple syntax error.

In summary, subjects appear to learn some new tasks by

applying previous knowledge and experience within and outside the domain of computers. In some cases, an old metaphor works well (word processing, bank machine). In other cases, the metaphor is extended to far and complications arise (icon/menu, calculators). In still other cases, the metaphor is inappropriate (graphs, tricks or games). The challenge is to examine, in more detail, with more subjects, the conditions under which metaphors contribute to and inhibit new learning. Examining the variety of metaphors used to solve new learning tasks can also provide information for educators about useful as well as troublesome metaphors.

Task Interpretation

While subjects are partially influenced by specific metaphors when learning new tasks, they are also influenced by their overall interpretation of a learning activity. Dweck (1986) brought this issue to the fore when he noted differences in students who had performance as opposed to learning goals. Pursuit of performance goals involved quick, task completion; pursuit of learning goals involved the attainment and maintenance of personally challenging knowledge.

Subjects in this study treated the spreadsheet exercises in the following ways: as tasks to be completed (performance goals), opportunities to build knowledge (learning goals), a situation in which they were being tested, a real-world work assignment, and an occasion to evaluate the software package.

Tasks and tests. Subjects who treated the exercise as tasks to be completed (3 and 5), also appeared to treat the exercises as a series of tests. Subject 1 openly expressed anxiety about being tested and joked about it on several occasions, even though he was instructed that the learning tasks were not tests. Subject 5 appeared to have a performance oriented approach to the learning tasks. She keyed rapidly, often unaware of the cause-and-effect of these keystrokes. She exhibited a desire to use shortcuts and "darted" around the spreadsheet menu line, again somewhat oblivious to the consequences of her actions. Bruner (1986) commented on this "darting" behaviour, calling it Vicarious Trial and Error (VTE) activity. He used VTE to study the effect of high intensity motivation on a rat attempting to negotiate it's way through a maze. The metaphor of a rat darting back and forth aptly captures the essence of high intensity, task focused behaviour.

Learning goals. Subjects 2 and 4, on the other hand took a much more leisurely and knowledge-directed approach to the learning tasks. Subject 2, for example, broadly explored the spreadsheet menu, carefully and slowly pressing the cursor keys, appearing to focus on the cause-and-effect of each key. She would display what I call "zooming in-zooming-out" behaviour, moving deeper into the menu and then backtracking.

Subject 4, was highly motivated to learn about the spreadsheet software package. He constantly probed about non-task

related information that might prove useful in the future; he would thank me for giving him hints and use those hints to acquire new knowledge on his own; he approached each exercise methodically and deliberately, making very few errors and capitalizing quickly on those that were made.

Real-world assignment. Subject 4 also approached the spreadsheet exercise as a real-world task by focusing on a number of aesthetic details and substantive concerns related to the kind of information he might have to enter in the future. It is unclear whether his desire to create a "good looking" spreadsheet that would be acceptable in a work environment contributed to his learning. It did prompt him to look at various formatting options which gave him specific knowledge about the kind of information that could be entered in a spreadsheet. No other subject acquired this kind of new information.

Evaluation. Subject 3's knowledge and experience of Macintosh software placed her in the role of being an evaluator. While the software was relatively unfamiliar to the other subjects, the computer was also unfamiliar to subject three. Instead of immersing herself in the task like subject 4, she distanced herself, and provided a running commentary on the drawbacks of the IBM-style, spreadsheet software. This drew her attention away from learning how to use a spreadsheet.

How a subject interprets a learning situation is related to how they acquire new knowledge. In this study, subjects were

readily identified by their overall learning agendas. These agendas affected the kinds of information they absorbed, the nature of exploration, the kinds of questions they asked, and the kinds of errors they made.

Use of Terminology

Vygotsky (1978) was a pioneer in exploring the role of language in thought. He noted that conceptual learning was a collaborative effort requiring supportive dialogue. Bruner (1986) added that conceptions of knowledge, thought, and learning are "fulfilled, impeded, and distorted by the way in which we talk about the world and think about it in the coin of that talk" (p. 121).

A primary aspect of learning in a particular environment is the language that individuals use to communicate to each other about commonly perceived actions and events (Bruner, 1983). As any novice will tell you, there is a definite (or indefinite) language used in a computer environment that has to be acquired in order to function effectively (Turkle, 1984). As well, there is a language, albeit somewhat restricted, that computer software uses to communicate to users. This also has to be learned, but is highly variable and dependent on the software used.

Subjects use of terminology in this study appeared to influence knowledge acquisition. At times, vague terminology, used by beginner and intermediate users, prevented them from exploring. Clear, but non-standard terminology, sometimes clashed

with appropriate "computerspeak" causing confusion. In certain cases, incorrect terminology precipitated extended difficulties.

Vague terminology. Subjects 3 and 5 had a vague notion that "commands" solve problems. When asked how they might pursue a particular task, they would often say "there would be a command". Commands were perceived as the panacea for all software problems. Unfortunately, this inhibited them from actively building their knowledge. A clearer understanding of the concept of commands, namely that they are interconnected and tend to appear in a extended sequences, might have altered the knowledge building process.

Sometimes subjects did not have terminology to describe an action or task. It was not that their terminology was vague--they simply had no words to describe their actions. Subject 3 exclaimed that "things were changing" when asked what was happening as she moved the cursor horizontally along the spreadsheet menu bar. Without a language to describe her actions, she was unable to notice the details of menu operation and seemed reluctant to explore further. This strongly influenced her future menu actions--she was often confused and disoriented when moving from the worksheet to the menu.

Non-standard terminology. Subject 4 appeared to have a clear understanding of how spreadsheet worked, but on occasion his terminology was non-standard, particularly when learning a new detail. He attempted to enter a telephone number into a cell

location e.g. 800-0400. Instead of showing the phone number, the software calculated the result of 800 minus 400. He eventually figured out that to enter a phone number, he needed to use a quotation mark first to indicate that he was not doing a mathematical calculation. He quickly coined the phrase "generic entry" to describe this action. He could use this phrase to communicate to himself, but would have difficulty when trying to talk with other spreadsheet users.

Incorrect terminology. One last example involves Subject 5. When entering the spreadsheet menu, the first command a user sees, along with several others, is "Worksheet". A worksheet is the place where all the numbers and labels are entered. When a user moves from the worksheet to the menu line she cannot enter numbers and labels--she must first exit the menu line. The command "Worksheet" refers to things that can be done to the worksheet--like widening columns and formatting numbers. This is a relatively difficult concept for beginners to learn, but Subjects 1 and 2 appeared to negotiate the task without too much difficulty. Subject 5, immediately began to call the menu line "the Worksheet" because that was the first word she saw. This mislabelling, brought her no end of trouble, both in communicating to herself and to me. This incorrect labelling also prevented her from noticing the difference between the worksheet and the menu line resulting in considerable confusion.

Terminology is clearly integrated with learning in a

computer environment. Vague terminology deters exploration and sometimes prevents the subject from forming a clear understanding of his or her actions. Non-standard terminology may be useful in communicating to oneself, but difficulties arise in a more populated setting. Incorrect terminology can cause heightened confusion. This brief protocol analysis illustrates the difficulty of learning a new language and how important that language is to adequate concept formation.

Summary

The purpose of this study was to examine protocol data of subjects who were learning a spreadsheet software package. Anecdotes were drawn focusing on four aspects of knowledge acquisition: previous computer knowledge, the use of metaphors, task interpretation, and terminology. Four conclusions were made. First, previous computer knowledge did not guarantee success and often inhibited learning. Second, subjects used a variety of metaphors to ground new knowledge; some proved quite useful, others impeded knowledge building. Third, a subject's global perception of a learning activity had considerable influence on success of task completion, pace of learning, and the kind of errors made. Forth, subjects use of computer terminology effected their progress through certain tasks. Vague words tended to inhibit exploration and complicate future actions. Subjects would also invent their own jargon or use appropriate words in

inappropriate settings. A final conclusion can also be made.

If researchers are to advance current theory in the area of computer abilities, they will have to alter their methods to explore the construction of knowledge while students are learning new concepts. Current Galtonian methods, while concise and easily understood, provide almost no information about the dynamics of human-computer interaction. The analysis of knowledge construction constraints from this study provides a preliminary understanding of how knowledge building occurs in a computer-environment. While the reliability and validity of the analysis is compromised by the small sample size and the examination of only one kind of computer with one kind of software, the use of videotaped, think-aloud protocols, is a promising example of the kind of process-oriented method that will help researchers understand how people acquire and use computer-related concepts.

References

Anderson, R. E., & Klassen, D. L. (1981). A conceptual framework for developing computer literacy instruction. AEDS Journal, 14(3), 128-143.

Bereiter, C., & Scardamalia, M. (1989). Intentional learning as a goal of instruction. In L. B. Resnick (Ed.) Knowing, Learning, and Instruction. Hillsdale, NJ: Erlbaum Associates (pp. 361-392).

Brown, A. L., & Palinscar, A. S. (1989). Guided, cooperative learning and individual knowledge acquisition. In L. B. Resnick (Ed.) Knowing, Learning, and Instruction. Hillsdale, NJ: Erlbaum Associates (pp. 393-451).

Bruner, J. (1983). Child's talk. Learning to use Language, Toronto: George J. McLeod Ltd.

Bruner, J. (1986). Actual minds, Possible Worlds. Cambridge, MA: Harvard University Press.

Ceci, S. J. (1990). On intelligence .. More or Less. A Bio-ecological Treatise on Intellectual Development. Prentice Hall: Englewood Cliffs, New Jersey.

Chi, M. T. H., & Bassok, M. (1989). Learning from examples via self-explanations. In L. B. Resnick (Ed.) Knowing, Learning, and Instruction. Hillsdale, NJ: Erlbaum Associates (pp. 251-282).

Collins, A. Brown, J. S., Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and

mathematics. In L. B. Resnick (Ed.) Knowing, Learning, and Instruction. Hillsdale, NJ: Erlbaum Associates (pp. 453-494).

Dweck, C. S. (1986). Motivational processes affecting learning. American Psychologist, 41(10), 1040-1048.

Ericsson, A. K., & Simon, H. A. (1984). Protocol Analysis: Verbal Reports as Data. MIT Press.

Glaser, R. (1990). The reemergence of learning theory within instructional research. American Psychologist, 45(1), 29-39.

Glaser, R., & Bassok, M. (1989). Learning theory and the study of instruction. Annual Review of Psychology, 40, 631-666.

Kato, T. (1986). What "question-asking protocols" can say about the suer interface. International Journal of Man-Machine Studies, 25, 659-673.

Kay, R. H. (1989). Bringing computer literacy into perspective. Journal of Research on Computing in Education, 22(1), 35-47.

Kay, R. H. (in press). The computer literacy potpourri: A review of the literature or McLuhan revisited. Journal of Research on Computing in Education

Keating, D. P. (in press). Charting pathways to the development of expertise. Educational Psychologist.

Kuhn, D. (1989). Children and adults as intuitive scientists. Psychological Review, 96(4), 674-689.

Moont, S. (1987). Computer literacy: An archaeological view. ECCO Output, 7(2), 14-17.

Oatley, K. (1988). Practical methods for evaluating human-

computer interfaces. Unpublished Manuscript.

Siegler, R. S. (1989). How domain-general and domain-specific knowledge interact to produce strategy choices. Merrill-Palmer Quarterly, 35(1), 1-26.

Sternberg, R. J. (1989). Domain-generality versus domain-specificity: The life and impending death of a false-dichotomy. Merrill-Palmer Quarterly, 35(1), 115-130.

Vygotsky, L. S. (1978). Mind in Society. Harvard University Press: Cambridge, MA.

APPENDIX A -PROTOCOL INSTRUCTIONS

1. What I'm going to ask you to do is to use this piece of software.

2. I'm trying to get some sense of how easy it is to use this program.

3. As you are interacting with the program, I'd like you to SAY OUT LOUD what it is you are trying to do.

   FOR EXAMPLE:

   a) if you are looking around say what it is you are looking at
   b) if you do anything say what it is your are doing
   c) If you get stuck, I'd like you to say what you think you need to do to get unstuck
   d) If you come across something unexpected, tell me about what you expected

4. Basically, say what's going on in your mind all of the time.

5. If you dry up, or get very absorbed in something, I may ask what you are thinking at the time.

6. This is not an intelligence test. In fact, it is not any kind of test at all.

PROMPTS

So you are going to ...
So what are you thinking about now ...
When you say "oh" ["oops"], something unexpected happened?
So you are going to try ...
So what happened here ?
So nothing seemed to happen?